In the United States Patent and Trademark Office

In re Application of:	**Stephen W. Edwards**
Examiner:	**Jeffery A. Brier**
Art Unit:	**2628**
Application No:	**09/353,887**
Filed:	**07/15/99**
Attorney Docket:	**TDH-29**
Title:	**Graphics Processor with Texture Memory Allocation System**

APPEAL BRIEF

Honorable Commissioner of Patents and Trademarks
PO Box 1450
Alexandria, VA 22313-1450

Sir:

Appellant herewith respectfully submits that Examiner Brier's final office action of 01/05/2007 and the advisory action of 04/19/2007, rejecting claims 1, 4-22 and 24-38, should be reversed in view of the following arguments and authorities.

Table of Contents

Real Party in Interest

The real party in interest in the present application and Appeal is 3Dlabs Inc. Ltd., a corporation of Bermuda, having a place of business at Huntsville, Alabama 35824. 3Dlabs Inc. Ltd. is a subsidiary of Creative Technologies, Ltd., a corporation of Singapore.

Related Appeals and Interferences

Attached is a copy of an "Order Returning Undocketed Appeal to Examiner" mailed March 8, 2006, by Dale M. Shaw, Program and Resource Administrator. Examiner Brier mailed his answer to that Remand on July 12, 2006, wherein the Applicant chose to reopen prosecution by a Request to Reopen Prosecution via Reply on September 12, 2007, followed by a Preliminary Amendment on October 4, 2006. In response to those submissions, Examiner Brier issued another Final Rejection on January 5, 2007. The Applicant filed a reply to the Final Rejection on April 5, 2007, to which Examiner Brier issued an Advisory Action on April 19, 2007. The Applicant filed a Notice of Appeal on July 5, 2007.

Status of Claims

Claims 1, 4-22 and 24-38 are pending and on appeal.

Status of Amendments

A response after Final Rejection (arguments only, no amendments to claims), which was filed on April 5, 2007 has been considered but has not been entered.

Table of Authorities

Patents Cited As References Against the Present Application

U.S. Patent No. 5,886,705 *"Texture Memory Organization Based on Data Locality"* to Derek J. ***Lentz*** (hereinafter "*Lentz*")

U.S. Patent No. 5,793,376 *"Method of Producing Image Data, Image Data Processing Apparatus, and Recording Medium"* to Masayoshi ***Tanaka***, Masaaki Oka, Teiji Yutaka, Kaoru Hagiwara, Hidetoshi Ichioka (hereinafter "*Tanaka*")

U.S. Patent No. 5,831,637 *"Video Stream Mixing for 3D Graphics Systems"* to David W. ***Young***, Jeffrey J. Holt, James Leroy Deming (hereinafter "*Young*")

U.S. Patent No. 6,046,747 *"Graphics Application Programming Interface Avoiding Repetitive Transfer of Texture Mapping Data"* to Bradley L. ***Saunders***, Brett E. Johnson (hereinafter "*Saunders*")

U.S. Patent No. 5,550,961 *"Image Processing Apparatus and Method of Controlling the Same"* to Hiroyuki ***Chimoto*** (hereinafter "*Chimoto*")

Cases

MPEP

Summary of Claimed Subject Matter

The following summary refers to disclosed embodiments and their advantages, but does not delimit any of the claimed inventions.

The present Application relates to processing graphics request data for display on a computer display device, and more specifically, to:

- graphics accelerators (Independent Claims 1, 21, 29)
- method for applying texture to a graphical image (Independent Claim 9)
- method of storing a texture map in a single linear texture memory of a graphics accelerator (Independent Claim 26)
- computer program product (Independent Claims 15, 32), and
- data structures for storing data relating to a texture map (Independent Claim 35).

BACKGROUND

Many conventional three dimensional graphics processing programs apply texture maps to graphical images using a texture processor.[1] A program typically determines the location and type of texture map (e.g., its dimensional type) in the texture memory. Once this information is determined, the program transmits a message to the texture processor with this information. Upon receipt of the message by the texture processor, the texture map is retrieved and applied to a graphical image. Transmitting the message to the texture processor requires bus bandwidth that preferably is utilized for transmitting other graphics request code.[2] Also, texture memory commonly is configured as linear memory (i.e., one dimensional). Many texture maps, however, are two and three dimensions. Storing a higher dimensioned

[1] See e.g. Page 1, Lines 6-10.
[2] See e.g. Page 1, Lines 15-16.

texture map in linear texture memory thus often results in an inefficient allocation of memory resources.[3]

The following summary of independent claims per the instructions of MPEP MPEP §1205.02 Appeal Brief Content (*v*) have the required figure reference numbers and also references to pages and line numbers in the Specification where the particular element is further described.

SUMMARY OF INDEPENDENT CLAIM 1

Claim 1 is for a graphics accelerator **106**[4] comprising: a single texture buffer **304A**[5], a plurality of texture processors **302A/302B**[6] retrieving texture packets **700**[7] from the single texture buffer, each texture processor including a fetching engine **308**[8] that retrieves (**step 602**[9]) the texture packets, each texture packet being stored (**step 410**[10]) in the texture buffer and being associated with a texture map (**FIG. 5**)[11] that is different than the texture maps associated with any other texture packet in the texture buffer, each texture packet including data relating to the location **710**[12] of its associated texture map in the texture buffer and data **704**[13] relating to the dimensional type of that texture packet's associated texture map; wherein the graphics accelerator is configured to convert (**step 404**[14]) the associated texture map to a one dimensional texture map if said dimensional type is greater than a one dimensional type by

---

[3] See e.g. Page 1, Line 18 to Page 2, Line 2.

[4] See e.g. Page 5, Lines 7-8, 15-28. Reference numbers merely illustrate one possible application of the claims.

[5] See e.g. Page 6, Lines 1-20 *passim*. Reference numbers merely illustrate one possible application of the claims.

[6] See e.g. Page 6, Lines 1-26 *passim*. Reference numbers merely illustrate one possible application of the claims.

[7] See e.g. Page 11, Line 28 through Page 12 Line 13. Reference numbers merely illustrate one possible application of the claims.

[8] See e.g. Page 6, Line 14. Reference numbers merely illustrate one possible application of the claims.

[9] See e.g. Page 11, Lines 8-10. Reference numbers merely illustrate one possible application of the claims.

[10] See e.g. Page 8, Lines 12-13. Reference numbers merely illustrate one possible application of the claims.

[11] See e.g. Page 4, Lines 21-22 *passim*. Reference numbers merely illustrate one possible application of the claims.

[12] See e.g. Page 12, Lines 8-13. Reference numbers merely illustrate one possible application of the claims.

[13] See e.g. Page 12, Lines 1-2. Reference numbers merely illustrate one possible application of the claims.

[14] See e.g. Page 7, Lines 12-13. Reference numbers merely illustrate one possible application of the claims.

defining a plurality of data blocks **1-20**[15] (in FIG.5) within the texture map and then assigning a sequence number to each of the data blocks; and wherein the consecutive data blocks of the texture map are stored consecutively in memory locations. (**step 408**[16])

SUMMARY OF INDEPENDENT CLAIM 9

A method of applying texture to a graphical image employing a graphics accelerator **106**[17] with a plurality of texture processors **302A/302B**[18], the method comprising: locating a texture packet **700**[19] identifying the location of a texture map **(FIG.5)**[20] in a single memory device **304A**[21], wherein the texture packet is associated with the texture map that is different than texture maps associated with other texture packets; parsing[22] the texture packet to determine the location of the texture map; retrieving (**step 608**[23]), based upon the determined location, the texture map from the single memory device; and applying[24] the texture map to the graphical image.

SUMMARY OF INDEPENDENT CLAIM 15

A computer program product[25] for use on a computer system **100**[26] with a plurality of texture processors **304A**[27] for applying texture to a graphical image, the

---

[15] See e.g. Page 7 Line 29 through Page 8 Line 3. Reference numbers merely illustrate one possible application of the claims.

[16] See e.g. Page 8, Lines 8-10. Reference numbers merely illustrate one possible application of the claims.

[17] See e.g. Page 5, Lines 7-8, 15-28. Reference numbers merely illustrate one possible application of the claims.

[18] See e.g. Page 6, Lines 1-26 *passim*. Reference numbers merely illustrate one possible application of the claims.

[19] See e.g. Page 11, Line 28 through Page 12 Line 13. Reference numbers merely illustrate one possible application of the claims.

[20] See e.g. Page 4, Lines 21-22 *passim*. Reference numbers merely illustrate one possible application of the claims.

[21] See e.g. Page 6, Lines 1-20 *passim*. Reference numbers merely illustrate one possible application of the claims.

[22] See e.g. Page 2, Lines 26-28. Reference numbers merely illustrate one possible application of the claims.

[23] See e.g. Page 11, Lines 19-20. Reference numbers merely illustrate one possible application of the claims.

[24] See e.g. Page 11, Lines 26-27. Reference numbers merely illustrate one possible application of the claims.

[25] See e.g. Page 12, Line 26. Reference numbers merely illustrate one possible application of the claims.

[26] See e.g. Page 4, Line 29. Reference numbers merely illustrate one possible application of the claims.

computer program product comprising a computer usable medium[28] having computer readable program code thereon, the computer readable program code including: program code for locating a texture packet identifying the location of a texture map in a single memory device, wherein the texture packet is associated with the texture map that is different than texture maps associated with other texture packets; program code for parsing the texture packet to determine the location and the number of dimensions of the texture map; program code for retrieving, based upon the determined location, the texture map from the memory device; and program code for applying the texture map to the graphical image.[29]

SUMMARY OF INDEPENDENT CLAIM 21

A graphics accelerator **106**[30] for processing a graphical image, the graphics accelerator comprising: a single texture buffer **304A**[31] for storing texture maps **(FIG. 5)**[32] and data relating to the texture maps stored in the texture buffer **710**[33]; and a plurality of texture processors **302A/302B**[34] that performs texturing operations on the graphical image, the plurality of the texture processors retrieving **(step 602**[35]**)** texture packets **700**[36] from the single texture buffer, each texture processor including a fetching engine **308**[37] that retrieves **(step 602**[38]**)** texture packets, each texture packet being stored in the texture buffer and being associated with a texture map that is

---

[27] See e.g. Page 6, Lines 1-20 *passim*. Reference numbers merely illustrate one possible application of the claims.
[28] See e.g. Page 12, Lines 26-30. Reference numbers merely illustrate one possible application of the claims.
[29] See e.g. Page 12, Lines 25-26. Reference numbers merely illustrate one possible application of the claims.
[30] See e.g. Page 5, Lines 7-8, 15-28. Reference numbers merely illustrate one possible application of the claims.
[31] See e.g. Page 6, Lines 1-20 *passim*. Reference numbers merely illustrate one possible application of the claims.
[32] See e.g. Page 4, Lines 21-22 *passim*. Reference numbers merely illustrate one possible application of the claims.
[33] See e.g. Page 12, Lines 8-13. Reference numbers merely illustrate one possible application of the claims.
[34] See e.g. Page 6, Lines 1-26 *passim*. Reference numbers merely illustrate one possible application of the claims.
[35] See e.g. Page 11, Lines 8-10. Reference numbers merely illustrate one possible application of the claims.
[36] See e.g. Page 11, Line 28 through Page 12 Line 13. Reference numbers merely illustrate one possible application of the claims.
[37] See e.g. Page 6, Line 14. Reference numbers merely illustrate one possible application of the claims.
[38] See e.g. Page 11, Lines 8-10. Reference numbers merely illustrate one possible application of the claims.

different than the texture maps associated with any other texture packet in the texture buffer, each texture packet including data **704**[39] relating to the dimensional type of its associated texture map.

SUMMARY OF INDEPENDENT CLAIM 26

A method of storing a texture map[40] (FIG. 4) in a single linear texture memory[41] of a graphics accelerator **106**[42], the method comprising: A. determining the dimension of the texture map (**step 402**[43]); B. converting (**step 404**[44]) the texture map to a one dimensional texture map if the dimension of the texture map is determined to be more than one dimensional, the one dimensional texture map having a first number of consecutive data blocks; C. locating a second number of consecutive memory locations in the single texture memory, the first number being equal to the second number; and D. storing (**step 408**[45]) the one dimensional texture map in the located memory locations in the single textured memory.

SUMMARY OF INDEPENDENT CLAIM 29

A graphics accelerator **106**[46] for processing graphical request code, the graphics accelerator comprising: a single linear texture memory **304A**[47] for storing texture maps; a plurality of texture processors **302A/302B**[48] that applies textures to items to be displayed, the plurality of the texture processors retrieving (**step 602**[49])

---

[39] See e.g. Page 12, Lines 1-2. Reference numbers merely illustrate one possible application of the claims.
[40] See e.g. Page 6, Lines 27-31. Reference numbers merely illustrate one possible application of the claims.
[41] See e.g. Page 8, Lines 30-31. Reference numbers merely illustrate one possible application of the claims.
[42] See e.g. Page 5, Lines 7-8, 15-28. Reference numbers merely illustrate one possible application of the claims.
[43] See e.g. Page 7, Lines 8-11. Reference numbers merely illustrate one possible application of the claims.
[44] See e.g. Page 7, Lines 12-13. Reference numbers merely illustrate one possible application of the claims.
[45] See e.g. Page 8, Lines 10-12. Reference numbers merely illustrate one possible application of the claims.
[46] See e.g. Page 5, Lines 7-8, 15-28. Reference numbers merely illustrate one possible application of the claims.
[47] See e.g. Page 6, Lines 1-20 *passim*. Reference numbers merely illustrate one possible application of the claims.
[48] See e.g. Page 6, Lines 1-26 *passim*. Reference numbers merely illustrate one possible application of the claims.
[49] See e.g. See e.g. Page 11, Lines 8-10. Reference numbers merely illustrate one possible application of the claims.

texture packets from the single texture memory, each texture processor including a texture map converter that converts texture maps having dimensions greater than one dimensional to a one dimensional texture map, each dimensional texture map having a first number of consecutive data blocks, the texture processor further including means for locating (**step 406**[50]) a second number of consecutive memory locations in the texture memory, the first number being equal to the second number; and means for storing (**step 408**[51]) the one dimensional texture map in the located memory locations in the single texture memory.

SUMMARY OF INDEPENDENT CLAIM 32

A computer program product[52] for use on a computer system **100**[53] for storing a texture map **(FIG. 5)**[54] in a single linear texture memory **304A**[55] of a graphics accelerator **106**[56], the computer program product comprising a computer usable medium[57] having computer readable program code thereon, the computer readable program code including program code for determining the dimension of the texture map; program code for converting the texture map to a one dimensional texture map if the dimension of the texture map is determined to be more than one dimensional, the one dimensional texture map having a first number of consecutive data blocks; program code for locating a second number of consecutive memory locations in the texture memory, the first number being equal to the second number; and program code for storing the one dimensional texture map in the located memory locations in the single texture memory.

---

[50] See e.g. Page 7, Lines 29-31. Reference numbers merely illustrate one possible application of the claims.
[51] See e.g. Page 8, Line 8-12. Reference numbers merely illustrate one possible application of the claims.
[52] See e.g. Page 12, Line 26. Reference numbers merely illustrate one possible application of the claims.
[53] See e.g. Page 4, Line 29. Reference numbers merely illustrate one possible application of the claims.
[54] See e.g. Page 4, Lines 21-22 *passim*. Reference numbers merely illustrate one possible application of the claims.
[55] See e.g. Page 6, Lines 1-20 *passim*. Reference numbers merely illustrate one possible application of the claims.
[56] See e.g. Page 5, Lines 7-8, 15-28. Reference numbers merely illustrate one possible application of the claims.
[57] See e.g. Page 12, Lines 26-30. Reference numbers merely illustrate one possible application of the claims.

SUMMARY OF INDEPENDENT CLAIM 35

A data structure **700**[58] for storing data relating to a texture map, the texture map having an associated dimension and being stored at a given location in a single memory device **304A**[59], the data structure comprising a location field **710**[60] identifying the given location in the memory device; and a dimension field **700**[61] identifying the dimension of the texture map **(FIG. 5)**[62].

[58] See e.g. Page 11 Line 28 through Page 12 Line 13. Reference numbers merely illustrate one possible application of the claims.

[59] See e.g. Page 6, Lines 1-20 *passim*. Reference numbers merely illustrate one possible application of the claims.

[60] See e.g. Page 12, Lines 8-13. Reference numbers merely illustrate one possible application of the claims.

[61] See e.g. Page 12, Lines 1-2. Reference numbers merely illustrate one possible application of the claims.

[62] See e.g. Page 4, Lines 21-22 *passim*. Reference numbers merely illustrate one possible application of the claims.

Grounds of Rejection to Be Reviewed on Appeal

I. Whether Claims 1 and 4-8 are obvious over *Lentz* (U.S. Pat. No. 5,886,705), *Tanaka et al* (U.S. Pat. No. 5,793,371), *Young et al* (U.S. Pat. No. 5,831,637), *Saunders et al* (U.S. Pat. No. 6,046,747) and *Chimoto* (U.S. Pat. No. 5,550,961).

II. Whether Claims 21-22 and 24-25 are obvious over *Lentz* (U.S. Pat. No. 5,886,705), *Young et al* (U.S. Pat. No. 5,831,637), *Tanaka et al* (U.S. Pat. No. 5,793,371), and *Saunders et al* (U.S. Pat. No. 6,046,747).

III. Whether Claims 9-13, 15-19, and 35-38 are obvious over *Lentz* (U.S. Pat. No. 5,886,705), *Tanaka et al* (U.S. Pat. No. 5,793,371), and *Saunders et al* (U.S. Pat. No. 6,046,747).

IV. Whether Claims 14, 20, 26-28, and 32-34 over *Lentz* (U.S. Pat. No. 5,886,705), *Tanaka et al* (U.S. Pat. No. 5,793,371), *Saunders et al* (U.S. Pat. No. 6,046,747), and *Chimoto* (U.S. Pat. No. 5,550,961).

V. Whether Claims 29-31 are obvious over *Lentz* (U.S. Pat. No. 5,886,705), *Tanaka et al* (U.S. Pat. No. 5,793,371), *Saunders et al* (U.S. Pat. No. 6,046,747), *Chimoto* (U.S. Pat. No. 5,550,961) and *Young et al* (U.S. Pat. No. 5,831,637).

Grouping of Claims

The claims on appeal do not stand or fall together, since they contain distinct recitations which are relevant to patentability and to the specific rejections stated. Each claim argued separately should be considered separately. Argument: The fact that the claims use different formulations and/or have been argued separately, shows that, if their patentability is not considered separately, any adverse decision would show that some limitations of some claims, and/or some arguments presented, had been unfairly ignored.

Arguments

This Application discloses an innovative graphics accelerator which enables separation of the address and selected attributes of a texture map from the command stream originating from a CPU in a computer. This innovation moves the handling of texture objects from the handling of a CPU-controlled software data structure to automatic handling within graphics hardware, such as a graphics accelerator. It therefore reduces command bandwidth by reducing texture state to a single pointer to a texture packet. Also, it abstracts the location of the texture map such that the command stream only needs to know the address of the descriptor.

Overall, none of the references relied on by Examiner Brier shows

1) a texture memory with two KINDS of items in it,

2) a texture memory which includes both maps and pointers to the maps, or

3) a texture memory which automatically relocates maps into sequential locations for most efficient output.

Specifically, Examiner Brier has not shown any prior art or a combination of such art which discloses a graphics accelerator which:

- uses a "texture packet" data structure containing at least one texture map address ***and*** the dimensional type of that texture map (see e.g. Claim 1),
- stores and retrieves such packets (see e.g. Claim 1),
- has a texture buffer which necessarily has texture maps ***and*** texture packets stored within it (see e.g. Claim 1),
- converts the texture maps to one dimensional maps if the maps were originally multi-dimensional (see e.g. Claim 1), or
- stores the converted maps consecutively in memory (see e.g. Claim 1).

The Applicant asserts that Examiner Brier has misinterpreted the cited references to allow him to "equate" features of those references to elements of the present innovations, which can then be combined into the present innovations. The specific misinterpretations are:

a. *Lentz's* texture memory addresses are not the texture packets of the present Application,
b. *Tanaka's* command packets are not the texture packets of the present Application, and
c. *Saunders'* "target parameter" in a display list is not the dimensional type in a texture packet of the present Application.

Finally, Examiner Brier has missed a key aspect of the unobviousness of the present Application. By creating, storing, and retrieving texture packets on-board the accelerator, the present innovations move away from a CPU-controlled data structure and allow for automatic handling of texture objects within hardware such as a graphics accelerator. With the benefit of the present Application, one skilled in the art would recognize that this would reduce texture map messaging and thus free-up precious bus bandwidth, per the implied problem to be solved at Page 1 Lines 11-16.

I. Whether Claims 1 and 4-8 are obvious over *Lentz* (U.S. Pat. No. 5,886,705), *Tanaka et al* (U.S. Pat. No. 5,793,371), *Young et al* (U.S. Pat. No. 5,831,637), *Saunders et al* (U.S. Pat. No. 6,046,747) and *Chimoto* (U.S. Pat. No. 5,550,961).

a. Equating *Lentz's* texture memory addresses to texture packets is improper because texture packets have a different structure and function.

The crux of Examiner Brier's argument is summarized in the 4/19/07 Advisory Action at Page 2 Lines 15-21:

> "Applicants arguments are not persuasive because the Final rejection at Page 4 ***equates*** the texture packets to texture memory addresses and because applicants claim 1 does not give the texture packet a function different than the memory addresses of *Lentz*." [Emphasis added].

The Applicant disagrees. Texture packets are claimed in Claim 1 as:

> "each texture packet including data related to the location of its associated texture map and data relating to the dimensional type of that packet's associated texture map;"

Thus, a texture packet contains a memory address but also at least the dimensional type of the associated texture map. Thus, the texture packet is not equal to a memory address. Further, a preferred embodiment of a texture packet is detailed in Figure 7 of the present Application (*see also* Page 11 Line 28 to Page 12 Line 14). The packet in Figure 7 is clearly not equal to a memory address.

Examiner Brier stated that the texture packet of Claim 1 does not have a different function than *Lentz's* memory addresses. The Applicant roundly disagrees. The texture packet of Claim 1 has the following functions that are clearly different than *Lentz's* memory addresses:

1. "each texture being stored in the texture buffer" (Claim 1 limitation). In *Lentz*, memory addresses are not stored in texture memory.
2. "texture processors retrieving texture packets" (Claim 1 limitation). In *Lentz*, memory addresses are not retrieved; they are computed as they are needed.
3. "each texture packet including data related to the location of its associated texture map and data relating to the dimensional type of that packet's associated texture map;" (Claim 1 limitation). This carrying or dual-payload function is different than a singular memory address.

b. Texture packets are created by texture processors on-board a graphics accelerator whereas *Tanaka's* command packets travel the main bus.

In rejecting Claim 1 at Page 6 Line 4-7 of the Final Office Action, the Examiner Brier interprets *Tanaka*:

> "The combination of Lentz and Young do not explicitly disclose that a texture packets identifying the location of a texture map. However,

> Tanaka et al clearly discloses that the packet data, which represents the storage location of a texture data/map."

Tanaka discloses at Col 2 Lines 37-41 a processing method which "allows the original geometric data of the object supplied to a terminal to be processed by a coordinate transforming device for producing a packet of data of a given format which is then transmitted to a rendering device for drawing." *Tanaka* further states that "GTE 61" is the coordinate transforming device. (Column 22 Line 6-7). Figure 1 of *Tanaka* shows GTE 61 connected solely to CPU 51. This would require the packets created by GTE 61 to be passed through or sent through CPU 51 and are thus command packets that travel the main bus "B" from CPU 51 to GPU 62 in *Tanaka's* Figure 1. Because *Tanaka's* command packet travels the main bus, it is not a texture packet and does not provide a benefit of the present Application, e.g. reduction of bus bandwidth. Thus, Examiner Bier improperly equated *Tanaka's* packets to texture packets.

c. ***Young* is devoid of any mention of texture packets or an equivalent being stored in memory.**

Although *Young* teaches multiple processors (which is a limitation of the present Application's Claim 1), *Young* does not appear to teach a texture packet is stored in the texture buffer; particularly, *Young* does not teach or suggest the texture packet of the present innovations (which are associated with a texture map and which include data relating to the location of the associated texture map in the texture buffer) being stored in the texture buffer. In-fact, *Young* does have texture memory (Figure 2, references 251a, 252a, 253a, 254a). However, *Young* simply describes the capability of his texture memory as "The texture memory is capable of storing several sets of mip-mapped textures for subsequent texture mapping." (Col 6 Line s 38-39).

d. **The "target parameter" of *Saunders'* bind texture call is not equal to the dimension type in texture packets because the "target parameter" is inserted into a display list, and not into texture memory.**

At Page 6 Paragraph 3 of the Final Office Action, Examiner Brier states:

> "However, in an analogous art (texture mapping), *Saunders* et al discloses that "the special bind texture call includes a target parameter that defines the dimension of the texture map and an ID number that identifies the display list texture object."

In the subsequent Advisory Action at Page 3 Lines 5-6, Examiner Brier further states:

> "Saunders as having a parameter that defines the dimension of the texture map and because the claim [Claim 1 of the present Application] does not claim a use for this parameter.

> First, the Applicant **does** teach a use for the parameter, as in Claim 1:
>
> "wherein the graphics accelerator is configured to convert the associated texture map to a one dimensional texture map if said dimensional type is greater than a one dimensional type..."

Second, Saunders teaches inserting the "target parameter" into a display list, and not a texture buffer. *Saunders* states, as cited by Examiner Brier at Col. 6, Lines 56-67:

> "The display list texture object list is used for quickly identifying optimized textures. In step 154, a special bind texture call that references the display list texture object is inserted into the display list. The special bind texture call includes a target parameter that defines the dimension of the texture map and an ID number that identifies the display list texture object. The effect of these operations is that, when the texture map corresponding to a glTexImage command is determined to be optimizable, a bind texture call is substituted for the

> glTexImage command or commands in the display list. The bind texture call references the display list texture object containing the required texture information." [Emphasis added.]

Claim 1 of the present Application states:

> "each texture packet including data relating to the location of its associated texture map ***in the texture buffer*** and data relating to the dimensional type of that texture packet's associated texture map" [Emphasis added].

It is respectfully submitted that Examiner Brier has selected an aspect of *Saunders*, taken it out of its context in *Saunders*, and combined it with other elements from the various references without motivation or suggestion from any of the references. Further, *Saunders* only teaches a call to get the dimensional data, and does not teach or suggest that the dimensional data is stored in a texture packet as described in the present innovations.

e. *Chimoto* does not teach storage of one dimensional texture maps *consecutively* in memory locations. Further, *Chimoto* (or *Lentz*) does not teach three dimensional or higher order texture map conversion.

In his Final Office Action of 1/5/2007 at Page 7 Paragraph 7., Examiner Brier cites *Chimoto* for disclosing converting a two-dimensional texture map into a one dimensional texture map. However Claim 1 of the present Application states:

> "graphics accelerator is configured to convert the associated texture map to a one dimensional texture map if said dimensional type is greater than a one dimensional type by defining a plurality of data blocks within the texture map and then assigning a sequence number to each of the data blocks; and wherein the ***consecutive*** data blocks of the texture map are stored consecutively in memory locations." [Emphasis added]

Examiner Brier has not shown that *Chimoto* discloses storage in consecutive memory locations. Thus, Examiner Brier has not made a complete argument for all limitations in the claims as being obvious. The Applicant cannot find any reference in *Chimoto* to the limitation that the texture data be stored within consecutive memory locations. And, nowhere in *Chimoto* is the disclosure that 3D and higher order texture maps can be expressed as one dimensional maps and are stored within consecutive memory locations.

Claim 4 claims dimensional types of texture maps of up to three dimensions. No reference is seen to teach, disclose, or suggest the use of three dimensional texture maps. Examiner Brier refers the Applicant to Col 1 Line 51 of *Lentz* to note the words "not necessarily two dimensional". Examiner Brier asserts that those words disclose by suggestion the conversion of 3D or higher texture maps into 1D maps. The Applicant disagrees. The words "not necessarily two dimensional" do not teach or suggest conversion of 3D or higher dimension texture maps into 1D maps.

f. **The rejection under 35 USC § 103(a) is not proper because the prior art has been misinterpreted to provide the elements to be combined to meet all of the limitations of the present Application's claims.**

As argued above, *Lentz, Tanaka, Young*, and *Saunders* have been misinterpreted because features of their disclosures have been improperly equated to elements in the present Application's claims. Thus, there is no combination of pre-existing elements from prior art to be made to achieve the inventions of the present Application. Simply put, *Lentz's* memory address are not texture packets, *Tanaka's* command packets are not texture packets, *Young's* texture memory does not store texture packets, *Saunder's* "target parameter" is not stored in texture memory, and *Chimoto's* storage is not consecutive. Thus, properly-equated

elements are not in the prior art to make a combination. Thus, no combination of prior art discloses each and every limitation of Claim 1 and Claims 4-8.

g. **One of ordinary skill in the art would not have made the combination of the five references proposed by Examiner Brier because the references themselves contain no suggestion –explicit or implicit – to make the proposed combination.**

The MPEP at § 2145 V. states,

> "reliance on a large number of references in a rejection does not, without more, weigh against the obviousness of the claimed invention. See *In re Gorman*, 933 F.2d 982, 18 USPQ2d 1885 (Fed. Cir. 1991)." [Emphasis added.]

This passage appears to note that a large number of references doesn't necessarily equal nonobviousness; however, it also suggests that the number of references cited can, when there is other evidence, weigh against the obviousness of an invention. This is explicitly stated in *Gorman* ("without more").

In the present case, Examiner Brier has selected elements from five different references to reject the present innovations. However, it is respectfully submitted that several or all of these references do not appear to teach what Examiner Brier suggests. These flaws in the interpretation of the references, which were discussed *supra*, combined with the sheer number of references that were combined, do in fact weigh against the obviousness of the claimed invention. Further, there is no teaching or suggestion in the art to make the very selective choices Examiner Brier has made from the various references in order to argue the claimed invention is obvious. *Gorman* itself discusses the limitations on combining references:

> "When it is necessary to select elements of various teachings in order to form the claimed invention, we ascertain whether there is any

> suggestion or motivation in the prior art to make the selection made by the applicant. *Interconnect Planning Corp. v. Feil*, 774 F.2d 1132, 1143, 227 U.S.P.Q. (BNA) 543, 551 (Fed. Cir. 1985). "'Obviousness can not be established by combining the teachings of the prior art to produce the claimed invention, absent some teaching, suggestion or incentive supporting the combination.'" *In re Bond*, 910 F.2d 831, 834, 15 U.S.P.Q.2D (BNA) 1566, 1568 (Fed. Cir. 1990) (quoting *Carella v. Starlight Archery and Pro Line Co.*, 804 F.2d 135, 140, 231 U.S.P.Q. (BNA) 644, 647 (Fed. Cir. 1986))."
>
> "The extent to which such suggestion must be explicit in, or may be fairly inferred from, the references, is decided on the facts of each case, in light of the prior art and its relationship to the applicant's invention. As in all determinations under 35 U.S.C. § 103, the decisionmaker must bring judgment to bear. It is impermissible, however, simply to engage in a hindsight reconstruction of the claimed invention, using the applicant's structure as a template and selecting elements from references to fill the gaps. *Interconnect Planning*, 774 F.2d at 1143, 227 U.S.P.Q. (BNA) at 551. The references themselves must provide some teaching whereby the applicant's combination would have been obvious." [Emphasis added.]

Applicant therefore respectfully submits that one of ordinary skill in the art, if confronted with the problem of reducing command bandwidth of texture maps, would not have been motivated to make the particular selections from the cited references--none of which actually solves the problem addressed by the present application, in the way it is solved by the present application. It is respectfully submitted that, without the present claims as a template, one of ordinary skill in the art would not have found the present innovations obvious, in light of the cited references.

In the Final Office Action of 1/05/07, Examiner Brier also mentions at Page 3 Par 2 the motivation for making the proposed combination, stating:

> "the motivation given by the examiner of more rapid processing is a goal of one skilled in the computer graphics field in order to better computer generated images."

The Applicant respectfully submits that stating a general goal (faster computing) is not a motivation to make the specific combination of elements, selected from the five different references that Examiner Brier asserts. "It is impermissible within the framework of section 103 to pick and choose from any one reference only so much of it as will support a given position, to the exclusion of other parts necessary to the full appreciation of what such reference fairly suggests to one of ordinary skill in the art." *In re Hedges*, 228 U.S.P.Q. 685, 687 (Fed. Cir. 1986). [Emphasis added.]

It is therefore respectfully submitted that Examiner Brier uses impermissible hindsight, relying on the present claims themselves as a template, in order to determine what elements from the various prior art references to select and combine in rejecting the claims. There is no teaching or suggestion in any of the references to make the proposed combination, whether explicit or implicit. Further, making the proposed combination would significantly modify the functioning of any of the cited references, so that they themselves would no longer function as described.

When such departure from the teaching of a reference is needed in making a combination for obviousness, it is respectfully submitted that the combination is not obvious, especially when elements must be selectively spliced together from no less than five different references, combined to form an invention that functions as none of the references do themselves.

In conclusion, the Applicant respectfully requests that the rejection of Claims 1 and 4-8 be reversed.

II. Whether Claims 21-22 and 24-25 are over *Lentz* (U.S. Pat. No. 5,886,705), *Young et al* (U.S. Pat. No. 5,831,637), *Tanaka et al* (U.S. Pat. No. 5,793,371), and *Saunders et al* (U.S. Pat. No. 6,046,747).

Claims 21, 22, 24 and 25 include the same limitations as Claim 1 except that the limitation of requiring the texture processor to convert higher order texture maps is not a part of the claim. Thus, selected arguments made in the arguments *supra* against the rejection of Claim 1 apply equally to the rejection of Claims 21 and 22, including the inapplicability of *Lentz* (I.)(a.), the inapplicability of *Tanaka* (I.)(b.), the shortcoming of *Young* not requiring storage of texture packets in texture memory (I.)(c.), the inapplicability of *Saunders* (I.)(d.), and the inapplicability of 35 USC 103(a) because the cited references do not provide disclosure of the elements required to even make a combination as made in argument (I.)(f.). More specifically, *Lentz's* memory address are not texture packets, *Tanaka's* command packets are not texture packets, *Young's* texture memory does not store texture packets, and *Saunder's* "target parameter" is not stored in texture memory. Thus, properly-equated elements are not in the prior art to make a combination. Thus, no combination of prior art discloses each and every limitation of Claims 21, 22, 24 and 25.

In conclusion, the Applicant respectfully requests that the rejection of Claims 21, 22, 24 and 25 be reversed.

III. Whether Claims 9-13, 15-19, and 35-38 are obvious over *Lentz* (U.S. Pat. No. 5,886,705), *Tanaka et al* (U.S. Pat. No. 5,793,371), and *Saunders et al* (U.S. Pat. No. 6,046,747).

For the reasons cited supra, the *Lentz, Tanaka*, and *Saunders* references are inapplicable to the differentiating elements of Claims 9-13 and 15-19. More

specifically, *Lentz's* memory addresses are not texture packets, *Tanaka's* command packets are not texture packets, and . Thus, properly-equated elements are not in the prior art to make a combination.

For the reasons cited supra, the references cited are inapplicable to the differentiating elements of Claims 35-38. More specifically, *Lentz's* memory address are not texture packets, *Tanaka's* command packets are not texture packets, and *Saunder's* "target parameter" is not stored in texture memory. Thus, properly-equated elements are not in the prior art to make a combination.

In conclusion, the Applicant respectfully requests that the rejection of this group of claims be reversed.

IV. Whether Claims 14, 20, 26-28, and 32-34 are over *Lentz* (U.S. Pat. No. 5,886,705), *Tanaka et al* (U.S. Pat. No. 5,793,371), *Saunders et al* (U.S. Pat. No. 6,046,747), and *Chimoto* (U.S. Pat. No. 5,550,961).

For the reasons cited *supra*, the references cited are inapplicable to the differentiating elements of Claims 35-38. More specifically, *Lentz's* memory address are not texture packets, *Tanaka's* command packets are not texture packets, *Saunder's* "target parameter" is not stored in texture memory, and *Chimoto* does not require consecutive storage in texture memory. Thus, properly-equated elements are not in the prior art to make a combination.

In conclusion, the Applicant respectfully requests that the rejection of this group of claims be reversed.

V. Whether Claims 29-31 are over *Lentz* (U.S. Pat. No. 5,886,705), *Tanaka et al* (U.S. Pat. No. 5,793,371), *Saunders et al* (U.S. Pat. No. 6,046,747), *Chimoto* (U.S. Pat. No. 5,550,961) and *Young et al* (U.S. Pat. No. 5,831,637).

For the reasons cited *supra*, the references cited are inapplicable to the differentiating elements of Claims 35-38. More specifically, *Lentz's* memory address are not texture packets, *Tanaka's* command packets are not texture packets, *Young's* texture memory does not store texture packets, *Saunder's* "target parameter" is not stored in texture memory, and *Chimoto* does not require consecutive storage in texture memory. Thus, properly-equated elements are not in the prior art to make a combination.

In conclusion, the Applicant respectfully requests that the rejection of this group of claims be reversed.

Requested Relief

For the reasons advanced above, Appellant respectfully contends that all claims are patentable. Therefore, reversal of the rejections is respectfully requested.

To the extent necessary, a petition for an extension of time under 37 C.F.R. 1.136 is hereby made. Please charge any shortage in fees due in connection of this paper, including extension of time fees, to Deposit Account 07-2320 and please credit any excess fees to such deposit account.

October 5, 2007

Respectfully submitted,

Thomas J. Novak, Reg. No. 57,683
Patent Agent for the Applicant

Customer Number 29106
Groover & Holmes
PO Box 802889
Dallas, TX 75380
Tel: 972-980-5840
fax: 972-980 -5841

APPENDIX A – Text of Claims on Appeal

1. (Previously Presented) A graphics accelerator for processing a graphical image, the graphics accelerator comprising:

a single texture buffer for storing texture maps and data relating to the texture maps stored in the texture buffer; and

a plurality of texture processors that perform texturing operations on the graphical image, the plurality of the texture processors retrieving texture packets from the single texture buffer, each texture processor including a fetching engine that retrieves the texture packets,

each texture packet being stored in the texture buffer and being associated with a texture map that is different than the texture maps associated with any other texture packet in the texture buffer,

each texture packet including data relating to the location of its associated texture map in the texture buffer and data relating to the dimensional type of that texture packet's associated texture map;

wherein the graphics accelerator is configured to convert the associated texture map to a one dimensional texture map if said dimensional type is greater than a one dimensional type by defining a plurality of data blocks within the texture map and then assigning a sequence number to each of the data blocks; and wherein the consecutive data blocks of the texture map are stored consecutively in memory locations.

2-3. (Cancelled)

4. (Previously Presented) The graphics accelerator as defined by claim 1 wherein the dimensional type of each texture map is one of a one dimensional texture map, a two dimensional texture map, and a three dimensional texture map.

5. (Previously Presented) The graphics accelerator as defined by claim 1 wherein the texture processor further includes:

an input for receiving a texture message indicating that a texture map is to be utilized by the texture processor, the fetching engine responsively retrieving selected texture packets from the single texture buffer in response to receipt of the texture message.

6. (Original) The graphics accelerator as defined by claim 5 wherein the texture processor further includes:

a parsing engine for parsing a fetched texture packet and determining information relating to the texture map associated with the fetched texture packet.

7. (Original) The graphics accelerator as defined by claim 6 wherein the information relates to the location in the texture buffer of the texture map associated with the fetched texture packet.

8. (Original) The graphics accelerator as defined by claim 6 wherein the information relates to the number of dimensions of the texture map associated with the fetched texture packet.

9. (Previously Presented) A method of applying texture to a graphical image employing a graphics accelerator with a plurality of texture processors, the method comprising:

locating a texture packet identifying the location of a texture map in a single memory device, wherein the texture packet is associated with the texture map that is different than texture maps associated with other texture packets;

parsing the texture packet to determine the location of the texture map;

retrieving, based upon the determined location, the texture map from the single memory device; and

applying the texture map to the graphical image.

10. (Original) The method as defined by claim 9 wherein the texture packet is located by accessing a record identifying the location of the texture packet.

11. (Previously Presented) The method as defined by claim 9 wherein the single memory device is texture memory.

12. (Previously Presented) The method as defined by claim 9 wherein the texture packet is stored in the single memory device

13. (Previously Presented) The method as defined by claim 9 further comprising reconstructing the texture map after it is retrieved from the single memory device.

14. (Original) The method as defined by claim 13 wherein the texture packet includes data relating to the dimensional type of the texture map, the texture map being reconstructed by parsing the texture packet to determine the dimensional type of the texture map, the texture map being reconstructed based upon the determined dimensional type of the texture map.

15. (Previously Presented) A computer program product for use on a computer system with a plurality of texture processors for applying texture to a graphical image, the computer program product comprising a computer usable medium having computer readable program code thereon, the computer readable program code including:

program code for locating a texture packet identifying the location of a texture map in a single memory device, wherein the texture packet is associated with the texture map that is different than texture maps associated with other texture packets;

program code for parsing the texture packet to determine the location and the number of dimensions of the texture map;

program code for retrieving, based upon the determined location, the texture map from the memory device; and

program code for applying the texture map to the graphical image.

16. (Original) The computer program product as defined by claim 15 wherein the program code for locating includes program code for accessing a record identifying the location of the texture packet.

17. (Previously Presented) The computer program product as defined by claim 15 wherein the single memory device is texture memory.

18. (Previously Presented) The computer program product as defined by claim 15 wherein the texture packet is stored in the single memory device.

19. (Previously Presented) The computer program product as defined by claim 15 further comprising: program code for reconstructing the texture map after it is retrieved from the single memory device.

20. (Original) The computer program product as defined by claim 19 wherein the texture packet includes data relating to the dimensional type of the texture map, the program code for reconstructing comprising:

    program code for parsing the texture packet to determine the dimensional type of the texture map, the texture map being reconstructed based upon the determined dimensional type of the texture map.

21. (Previously Presented) A graphics accelerator for processing a graphical image, the graphics accelerator comprising:

a single texture buffer for storing texture maps and data relating to the texture maps stored in the texture buffer; and

a plurality of texture processors that performs texturing operations on the graphical image, the plurality of the texture processors retrieving texture packets from the single texture buffer, each texture processor including a fetching engine that retrieves texture packets, each texture packet being stored in the texture buffer and being associated with a texture map that is different than the texture maps associated with any other texture packet in the texture buffer, each texture packet including data relating to the dimensional type of its associated texture map.

22. (Previously Presented) The graphics accelerator as defined by claim 21 wherein each texture packet includes data relating to the location of its associated texture map in the single texture buffer.

23. (Cancelled)

24. (Original) The graphics accelerator as defined by claim 21 wherein the texture processor further includes:

an input for receiving a texture message indicating that a texture map is to be utilized by the texture processor, the fetching engine retrieving selected texture packets from the texture buffer in response to receipt of the texture message.

25. (Original) The graphics accelerator as defined by claim 24 wherein the texture processor further includes:

a parsing engine that parses a fetched texture packet and determines information relating to the texture map associated with the fetched texture packet.

26. (Previously Presented) A method of storing a texture map in a single linear texture memory of a graphics accelerator, the method comprising:

A. determining the dimension of the texture map;

B. converting the texture map to a one dimensional texture map if the dimension of the texture map is determined to be more than one dimensional, the one dimensional texture map having a first number of consecutive data blocks;

C. locating a second number of consecutive memory locations in the single texture memory, the first number being equal to the second number; and

D. storing the one dimensional texture map in the located memory locations in the single textured memory.

27. (Original) The method as defined by claim 26 wherein the texture map is two dimensional, step B comprising:

B1. defining a plurality of data blocks within the texture map; and

B2. assigning a sequence number to each of the data blocks, the sequence numbers being consecutive numbers.

28. (Original) The method as defined by claim 26 wherein step D comprises:

D1. consecutively storing each consecutive data block of the one dimensional texture map in the located memory locations.

29. (Previously Presented) A graphics accelerator for processing graphical request code, the graphics accelerator comprising:

a single linear texture memory for storing texture maps;

a plurality of texture processors that applies textures to items to be displayed, the plurality of the texture processors retrieving texture packets from the single texture memory, each texture processor including a texture map converter that converts texture maps having dimensions greater than one dimensional to a one dimensional texture map, each dimensional texture map having a first number of consecutive data blocks, the texture processor further including means for locating a second number of consecutive memory locations in the texture memory, the first number being equal to the second number; and

means for storing the one dimensional texture map in the located memory locations in the single texture memory.

30. (Original) The graphics accelerator as defined by claim 29 wherein the texture map converter comprises:

means for defining a plurality of data blocks within the texture map; and

means for assigning a sequence number to each of the data blocks, the sequence numbers being consecutive numbers.

31. (Original) The graphics accelerator as defined by claim 29 the storing means comprises: means for consecutively storing each consecutive data block of the one dimensional texture map in the located memory locations.

32. (Previously Presented) A computer program product for use on a computer system for storing a texture map in a single linear texture memory of a graphics accelerator, the computer program product comprising a computer usable medium having computer readable program code thereon, the computer readable program code including

program code for determining the dimension of the texture map;

program code for converting the texture map to a one dimensional texture map if the dimension of the texture map is determined to be more than one dimensional, the one dimensional texture map having a first number of consecutive data blocks;

program code for locating a second number of consecutive memory locations in the texture memory, the first number being equal to the second number; and

program code for storing the one dimensional texture map in the located memory locations in the single texture memory.

33. (Original) The computer program product as defined by claim 32 wherein the texture map is two dimensional, the program code for converting comprising:

program code for defining a plurality of data blocks within the texture map; and

program code for assigning a sequence number to each of the data blocks, the sequence numbers being consecutive numbers.

34. (Original) The computer program product as defined by claim 32 wherein the program code for storing comprises

program code for consecutively storing each consecutive data block of the one dimensional texture map in the located memory locations.

35. (Previously Presented) A data structure for storing data relating to a texture map, the texture map having an associated dimension and being stored at a given location in a single memory device, the data structure comprising

a location field identifying the given location in the memory device; and a dimension field identifying the dimension of the texture map.

36. (Original) The data structure as defined by claim 35 wherein the texture map comprises a set of mip maps, further wherein the location field includes a plurality of subfields, each subfield identifying the location of one mama in the set of mip maps.

37. (Previously Presented) The data structure as defined by claim 35 wherein the texture map spans a plurality of addresses in the single memory device, the location field identifying the plurality of addresses.

38. (Previously Presented) The data structure as defined by claim 35 wherein the data structure is stored in the single memory device, the single memory device being texture memory.

APPENDIX B – Application Drawings

The following drawings are draftsman duplications of the 7 partially hand-drawn figures included in the original application. These draftsman duplications are easier to refer to than the hand-drawn versions, and include no amendments or new matter.

Figure 1 schematically shows a portion of an exemplary computer system on which preferred embodiments of the invention may be implemented .

Figure 2 schematically shows a preferred graphics accelerator that may be utilized in accord with preferred embodiments of the invention.

Figure 3 shows additional details of a preferred embodiment of a rasterization stage shown in figure 2.

Figure 4 shows a preferred process of storing texture maps in either of the texture memories shown in figure 3.

Figure 5 shows an exemplary two-dimensional texture map as it is converted into a one dimensional texture map .

Figure 6 shows a preferred method of retrieving a texture map from texture memory.

Figure 7 schematically shows a texture packet configured in accord with preferred embodiments of the invention.

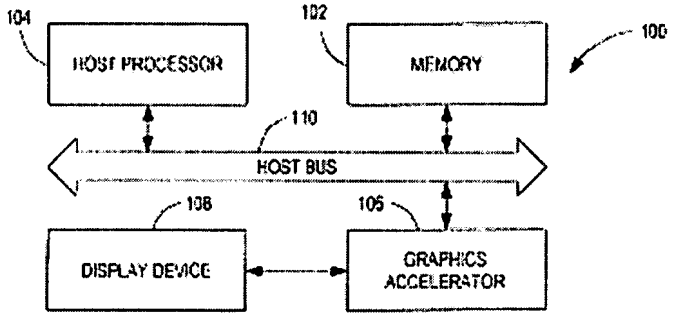

FIG. 1

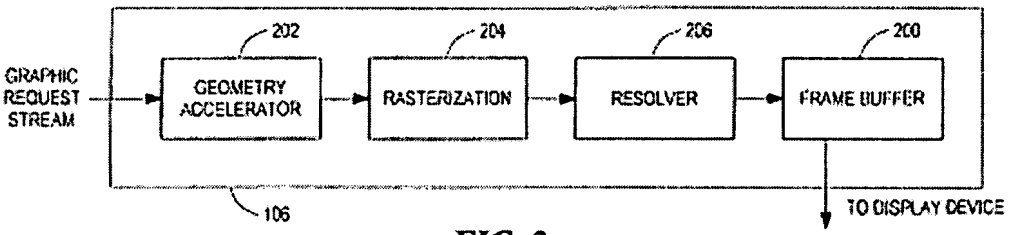

FIG. 2

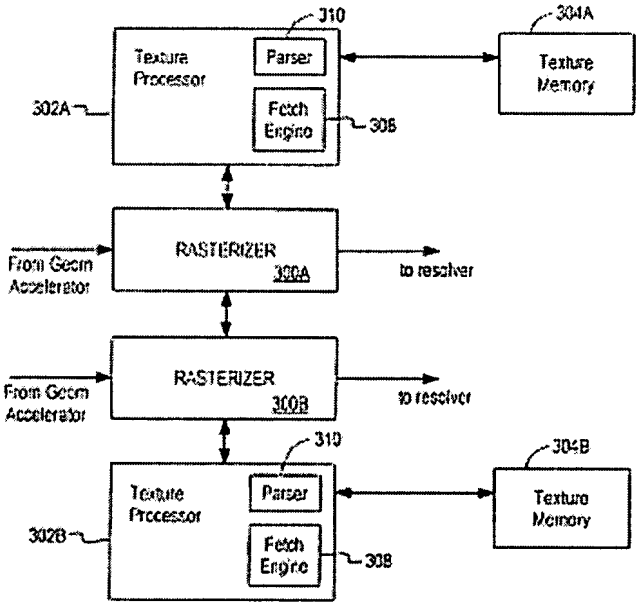

FIG. 3

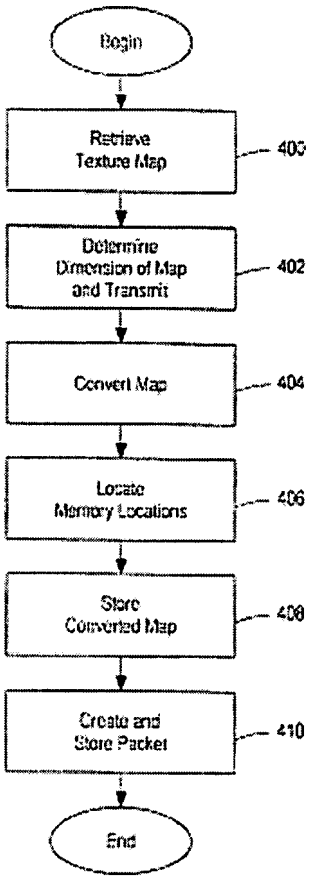
Begin
Retrieve Texture Map
400
Determine Dimension of Map and Transmit
402
Convert Map
404
Locate Memory Locations
406
Store Converted Map
408
Create and Store Packet
410
End

FIG. 4

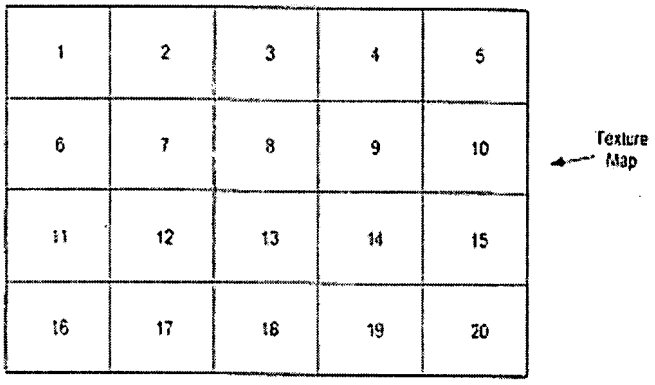
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
Texture Map

FIG. 5

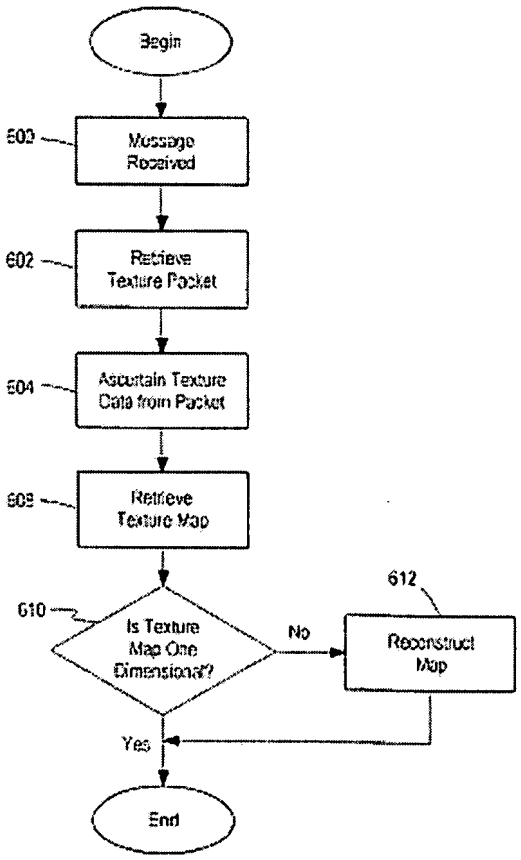
Begin
600
Message Received
602
Retrieve Texture Packet
604
Ascertain Texture Data from Packet
608
Retrieve Texture Map
610
Is Texture Map One Dimensional?
No
612
Reconstruct Map
Yes
End

FIG. 6

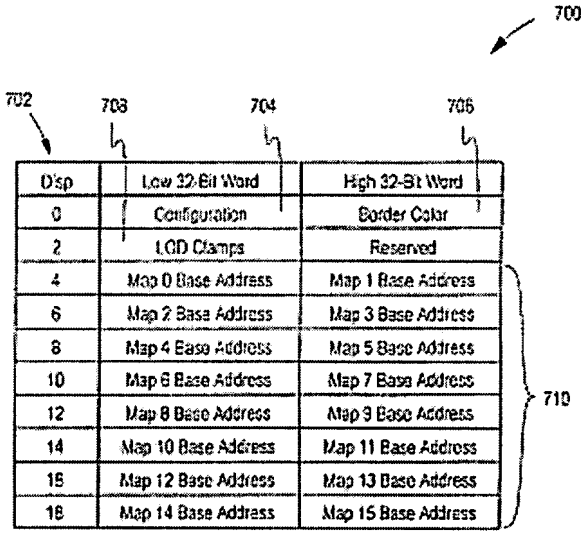
700
702
703
704
705
Disp
Low 32-Bit Word
High 32-Bit Word
0
Configuration
Border Color
2
LOD Clamps
Reserved
4
Map 0 Base Address
Map 1 Base Address
6
Map 2 Base Address
Map 3 Base Address
8
Map 4 Base Address
Map 5 Base Address
10
Map 6 Base Address
Map 7 Base Address
12
Map 8 Base Address
Map 9 Base Address
14
Map 10 Base Address
Map 11 Base Address
16
Map 12 Base Address
Map 13 Base Address
18
Map 14 Base Address
Map 15 Base Address
710

FIG. 7

APPENDIX C – Copy of Notice of Appeal

Attached

Applicant: Stephen W. Edwards
Title: Graphics Processor with Texture Memory Allocation System
Docket No.: TDH-29

Certificate under 37 CFR 1.10 of Mailing by "Express Mail"

EV 887057630 US
"Express Mail Label Number

October 5, 2007
Date of Deposit

I hereby certify that this correspondence is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to: MS Patent Application, Commissioner for Patents, PO Box 1450, Alexandria VA 22313-1450.

Carol Boultinghouse
Signature of person mailing correspondence

Carol Boultinghouse
Typed or printed name of person mailing correspondence

Enclosures:

1. Transmittal (1 page)
2. Fee Transmittal (1 page)
3. Petition for Extension of Time (1 page)
4. Appeal Brief (28 pages)
5. Appendix A –Text of Claims on Appeal (17 pages)
6. Appendix B – Application Drawings (3 pages)
7. Appendix C – Copy of Notice of Appeal (1 page)
8. Appendix D – Evidence (1 page)
9. Appendix E – Related Proceedings (7 pages)
10. Copy of Notice of Appeal (1 page)
11. Express Mail Certificate (1 pg)
12. Two (2) Return Post Cards (2 pages)

APPENDIX D: Evidence

NONE

APPENDIX E: Related Proceedings

Attached

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte STEPHEN W. EDWARDS

Application 09/353,887

ORDER RETURNING UNDOCKETED APPEAL TO EXAMINER

This application was received electronically at the Board of Patent Appeals and Interferences on January 27, 2006. A review of the application has revealed that the application is not ready for docketing as an appeal. Accordingly, the application is herewith being electronically returned to the examiner. The matters requiring attention prior to docketing are identified below:

A review of the Image File Wrapper (IFW) indicates that appellant filed a Final Rejection was filed on January 12, 2004 which lists the following rejections:

> 1. Claims 1, 4-13, 15-19, 21-22, 24-25 and 35-38 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lentz (5,886,705) in view of Young et al (U.S 5,831,637) and Tanaka et al (5,793,371), and further in view of Saunders et al (6,046,747) [page 2]; and

pg 52

2. Claims 14, 20 and 26-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lentz, [Y]oung and Tanaka et al in view of Saunders et al, and further in view of Chimoto (5,550,961) [page 13].

However, in the Examiner's Answer mailed May 5, 2005, the following rejections are listed:

1. Claims 1, 4-8, 21-22, and 24-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lentz (U.S. Pat. No. 5,886,705) in view of Young et al (U.S. Pat. No. 5,831,637) and Tanaka et al (U.S. Pat. No. 5,793,371), and further in view of Saunders et al (U.S. Pat. No. 6,046,747) [pages 4 and 5];

2. Claims 9-13, 15-19 and 35-38 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lentz (U.S. Pat. No. 5,886,705) in view of Tanaka et al (U.S. Pat. No. 5,793,371), and further in view of Saunders et al (U.S. Pat. No. 6,046,747) [page 10];

3. Claims 14, 20, 26-28 and 32-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lentz (U.S. Pat. No. 5,886,705) and Tanaka et al (U.S. Pat. No. 5,793,371) in view of Saunders et al (U.S. Pat. No. 6,046,747), and further in view of Chimoto (U.S. Pat. No. 5,550,961) [page 15]; and

4. Claims 29-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lentz (U.S. Pat. No. 5,886,705), Tanaka et al (U.S. Pat. No. 5,793,371) and Saunders et al (U.S. Pat. No. 6,046,747) in view of Chimoto

pg 53

(U.S. Pat. No. 5,550,961), and further in view of Young et al (U.S. Pat. No. 5,831,637) [page 19].

It appears that the four rejections appearing in the Examiner's Answer are new grounds of rejection.

37 CFR § 41.39 reads as follows:

§ 41.39 Examiner's answer.

(a)(1) The primary examiner may, within such time as may be directed by the Director, furnish a written answer to the appeal brief including such explanation of the invention claimed and of the references relied upon and grounds of rejection as may be necessary, supplying a copy to appellant. If the primary examiner determines that the appeal does not comply with the provisions of §§ 41.31 and 41.37 or does not relate to an appealable action, the primary examiner shall make such determination of record.

(2) An examiner's answer may include a new ground of rejection.

(b) If an examiner's answer contains a rejection designated as a new ground of rejection, appellant must within two months from the date of the examiner's answer exercise one of the following two options to avoid sua sponte dismissal of the appeal as to the claims subject to the new ground of rejection:

(1) Reopen prosecution. Request that prosecution be reopened before the primary examiner by filing a reply under § 1.111 of this title with or without amendment or submission of affidavits (§§ 1.130, 1.131 or 1.132 of this title) or other evidence. Any

pg 54

> amendment or submission of affidavits or other evidence must berelevant to the new ground of rejection. A request that complies with this paragraph will be entered and the application or the patent under ex parte reexamination will be reconsidered by the examiner under the provisions of § 1.112 of this title. Any request that prosecution be reopened under this paragraph will be treated as a request to withdraw the appeal.
>
> (2) Maintain appeal. Request that the appeal be maintained by filing a reply brief as set forth in § 41.41. Such a reply brief must address each new ground of rejection as set forth in § 41.37(c)(1)(vii) and should follow the other requirements of a brief as set forth in § 41.37(c). A reply brief may not be accompanied by any amendment, affidavit (§§ 1.130, 1.131 or 1.132 of this title) or other evidence. If a reply brief filed pursuant to this section is accompanied by any amendment, affidavit or other evidence, it shall be treated as a request that prosecution be reopened before the primary examiner under paragraph (b)(1) of this section.
>
> (c) Extensions of time under § 1.136(a) of this title for patent applications are not applicable to the time period set forth in this section. See § 1.136(b) of this title for extensions of time to reply for patent applications and § 1.550(c) of this title for extensions of time to reply for ex parte reexamination proceedings.

In order to include a new ground of rejection in the Examiner's Answer, the examiner must follow the guidelines set forth in training material entitled "Rules of Practice Before the Board of

Patent Appeals and Interferences, Final Rule," located at the following URL:

www.uspto.gov/web/offices/dcom/bpai/fr2004/moreinfo.html

The requirements for a new ground of rejection are:

1) Approval by a Technology Center Director or designee; and

2) Prominently identified, by a separate heading with all capital letters in the following sections of the Examiner's Answer:

Grounds of Rejection to be Reviewed on Appeal section, and

Grounds of Rejection section.

To correct this problem, the examiner will need to vacate the Examiner's Answer mailed May 5, 2005. Once the Examiner's Answer mailed May 5, 2005 is vacated, the examiner has the following options:

1) to write a new Examiner's Answer without the new grounds of rejection;

2) to reopen prosecution; or

3) to write a new Examiner's Answer properly setting forth the new grounds of rejection.

Accordingly, it is

ORDERED that the application is returned to the examiner to:

1) vacate the Examiner's Answer mailed May 5, 2005;

2) to select one of the following options:

a) reopen prosecution;

b) write a new Examiner's Answer without the new grounds of rejection; or

c) write a new Examiner's Answer properly setting forth the new grounds of rejection; and

3) for such further action as may be appropriate.

BOARD OF PATENT APPEALS
AND INTERFERENCES

By: _____

DALE M. SHAW
Program and Resource Administrator
(571) 272-9797

DMS:psb

Carlton Fields, PA
1201 West Peachtree Street
3000 One Atlantic Center
Atlanta, GA 30309

AF

PTO/SB/31 (04-07)
Approved for use through 09/30/2007. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

NOTICE OF APPEAL FROM THE EXAMINER TO THE BOARD OF PATENT APPEALS AND INTERFERENCES

Docket Number (Optional): TDH-29

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)]
on _____
Signature: Sue Wathers
Typed or printed name: Sue Wathers

In re Application of

Application Number	Filed
09/353,887	7/15/1999

For Graphic Processor w/Texture Memory allocation

Art Unit	Examiner
2628	Jeffery A. Brier

Applicant hereby appeals to the Board of Patent Appeals and Interferences from the last decision of the examiner.

The fee for this Notice of Appeal is (37 CFR 41.20(b)(1)) $ 500.00

☐ Applicant claims small entity status. See 37 CFR 1.27. Therefore, the fee shown above is reduced by half, and the resulting fee is: $ _____

☐ A check in the amount of the fee is enclosed.

☐ Payment by credit card. Form PTO-2038 is attached.

☐ The Director has already been authorized to charge fees in this application to a Deposit Account. I have enclosed a duplicate copy of this sheet.

☒ The Director is hereby authorized to charge any fees which may be required, or credit any overpayment to Deposit Account No. 072320 . I have enclosed a duplicate copy of this sheet.

☒ A petition for an extension of time under 37 CFR 1.136(a) (PTO/SB/22) is enclosed.

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

I am the

☐ applicant/inventor.

☐ assignee of record of the entire interest. See 37 CFR 3.71. Statement under 37 CFR 3.73(b) is enclosed. (Form PTO/SB/96)

☐ attorney or agent of record. Registration number _____

☒ attorney or agent acting under 37 CFR 1.34. Registration number if acting under 37 CFR 1.34. 30059

Signature

Robert Groover
Typed or printed name

972-980-5840
Telephone number

July 5, 2007
Date

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

☒ *Total of 2 forms are submitted.

This collection of information is required by 37 CFR 41.31. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11, 1.14 and 41.6. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

07/10/2007 HDESTA1 00000064 072320 09353887

01 FC:1401 500.00 DA